Mixture of Online and Offline Experts for Non-Stationary Time Series

Zhilin Zhao[1,2,3], **Longbing Cao**[2], **Yuanyu Wan**[4]

[1] School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China
[2] School of Computing, Macquarie University, Sydney, Australia
[3] Key Laboratory of Machine Intelligence and Advanced Computing, Ministry of Education, China
[4] School of Software Technology, Zhejiang University, Ningbo, China
zhaozhl7@hotmail.com, longbing.cao@mq.edu.au, wanyy@zju.edu.cn

Abstract

We consider a general and realistic scenario involving non-stationary time series, consisting of several offline intervals with different distributions within a fixed offline time horizon, and an online interval that continuously receives new samples. For non-stationary time series, the data distribution in the current online interval may have appeared in previous offline intervals. We theoretically explore the feasibility of applying knowledge from offline intervals to the current online interval. To this end, we propose the Mixture of Online and Offline Experts (MOOE). MOOE learns static offline experts from offline intervals and maintains a dynamic online expert for the current online interval. It then adaptively combines the offline and online experts using a meta expert to make predictions for the samples received in the online interval. Specifically, we focus on theoretical analysis, deriving parameter convergence, regret bounds, and generalization error bounds to prove the effectiveness of the algorithm.

Introduction

For a non-stationary time series, the data distribution in the current time window may have appeared in the past. Therefore, can we apply the knowledge from historical data to the current time window? In this paper, we theoretically prove that this is feasible.

A common assumption in statistical learning theories for time series is that observed samples are i.i.d., or stationary in stochastic processes (Hamilton 1994). To leverage sample dependence in non-i.i.d. processes, it is often assumed that observations come from a stationary ϕ-mixing or β-mixing sequence (Mohri and Rostamizadeh 2010). However, these assumptions may not hold as the distribution of real-life time series usually changes over time, making the hypothesis class not (agnostically) PAC learnable (Hanneke 2016). Fortunately, distribution changes in real life are often gradual, and samples in a short interval are nearly identically distributed (Kuznetsov and Mohri 2015). Therefore, we consider a realistic scenario involving non-stationary time series with several offline intervals of different distributions within a fixed offline time horizon and an online interval that continuously receives new samples. Once the number of received labeled samples reaches a predefined size, the online interval is converted to the last offline interval, and a new online interval begins.

Some existing methods (Shalev-Shwartz 2012; Zhang, Lu, and Zhou 2018) train an expert on the entire time series using off-the-shelf online optimization techniques, without considering the non-stationary nature of the data. However, the dynamic data with varying distributions can mislead the expert. Other methods (Yu 1994; Mohri and Rostamizadeh 2008, 2010) train an expert from scratch for each new online interval, which is safer but unreliable due to the scarcity of labeled samples at the early stage. Thus, it is fundamental yet highly challenging to design a learning method with tight sample complexity that outputs a hypothesis with desirable generalization. For non-stationary time series, the data distribution in the current online interval may have appeared in historical offline intervals. Therefore, a natural solution is to combine the offline experts from the offline intervals with the online expert from the current interval to address the shortcomings of the aforementioned methods.

Inspired by the mixture of experts (Puigcerver et al. 2024), we propose Mixture of Online and Offline Experts (MOOE) to transfer knowledge from offline intervals to the online interval, addressing the non-stationarity issue. Following the paradigm of prediction with expert advice (Cesa-Bianchi and Lugosi 2006; van Erven and Koolen 2016), MOOE employs a meta expert to combine the online and offline experts by adaptively weighting them according to their effectiveness. Specifically, the online expert is continuously updated in the current online interval using an existing online optimization method. Additionally, when an online interval collects enough samples to become an offline interval, all samples from this interval, along with previously obtained offline experts, are used to train the offline expert for this interval.

Theoretically, we prove that the regret of MOOE is determined by the regret of the off-the-shelf online optimization method used for the online expert. However, this can be improved if the number of maintained experts is within a bound controlled by the size of intervals and the empirical errors of the offline experts. By connecting optimization with learning theory (Hazan 2016), we derive the generalization error bound by jointly exploiting the regret, the properties of the loss function, the hypothesis class, and the data distribution, thereby verifying the effectiveness of our approach. Exper-

imentally, MOOE outperforms state-of-the-art methods for handling non-stationary time series.

Related Work

Learning Theory for Non-Stationary Time Series

For non-i.i.d. processes, under the stationary and β-mixing assumptions, the early work (Yu 1994) establishes the convergence rate over VC-dimension, and the work in (Mohri and Rostamizadeh 2008) presents data-dependent bounds in terms of the Rademacher complexity. By exploiting the stability properties of a specific learning algorithm, generalization bounds for ϕ-mixing and β-mixing sequences are provided in (Mohri and Rostamizadeh 2010). However, the mixing assumption is hard to be verified in practice. There are some attempts to relax the stationary and mixing assumptions. The uniform convergence under ergodicity sampling is shown in the work of (Adams and Nobel 2010). For an asymptotically stationary (mixing) process, although a generalization error is derived in (Agarwal and Duchi 2013) through the regret of an online algorithm, and their analysis depends on the assumption that the output from an online learning algorithm tends to be stable, which is invalid in a dynamic environment. In (Kuznetsov and Mohri 2014), the guarantee of the learning rate for nonstationary mixing processes is given by a sub-sample selection technique with the Rademacher complexity. Further, in (Kuznetsov and Mohri 2015), a more general scenario of nonstationary and non-mixing processes is considered, which proves the learning guarantees with the conditions of the discrepancies between distributions.

Regret Analysis in Dynamic Environments

The regret theory (Buchbinder et al. 2016) for measuring the performance has been extensively studied. The dynamic regret (Anagnostides, Farina, and Sandholm 2023) and its restricted form (Besbes, Gur, and Zeevi 2015) have been introduced to manage changing environments. A basic idea behind such regrets is to compare the cumulative loss of the learned expert with several experts rather than the best one. Along this line of study, adaptive learning for dynamic regret (Ader) (Zhang, Lu, and Zhou 2018) considers multiple experts with various learning rates updated by online gradient descent (OGD) (Zinkevich 2003), and the established upper bound matches the lower bound. Another independent work for dynamic regret in a nonstationary environment is about multi-armed bandit (MAB) (Besbes, Gur, and Zeevi 2015), where the work in (Wei, Hong, and Lu 2019) reveals how the statistical variance of the loss distributions affect the dynamic regret bound. However, these dynamic regrets depend on the distribution changing times, which are usually unknown. When the sequence of samples is very long, the data distribution may have changed many times. As a result, the loose bound cannot measure the learned expert performance in the current interval. Another limitation is that the bound is inappropriate for analyzing experts learned on the fly because these regrets only act on observed samples.

Problem Statement

In non-stationary time series, an online platform containing experts will receive an input $\mathbf{x}$ at each time step and predict its label y, indicating the class the input belongs to. For inputs with feedback, i.e., when the ground truth labels are revealed after predicting, the online platform will update by learning from the feedback. For inputs without feedback, the online platform merely predicts the labels. We aim to continuously update and utilize the experts in this online platform to more accurately predict class labels for samples from non-stationary time series in the current online interval.

Specifically, the considered non-stationary time series contains $G-1$ offline intervals and one online interval. Each offline interval contains B samples, and the online interval contains T samples ($T \in [B]$). We assume the distribution changes gradually, and the samples in each interval can be approximately drawn from a distribution. Accordingly, we set the maximal sample size B as a hyperparameter, even if the time between distribution changes is not constant and usually unknown. The online interval will become offline if $T = B$, increasing the number of offline intervals. Accordingly, we have the following assumptions.

Assumption 1 *Let $\mathcal{D}_g$ be the data distribution in the g^{th} interval. $\mathcal{D}_{\mathcal{U}} = \bigcup_{g=1}^{G} \mathcal{D}_g$ is non-stationary since $\mathcal{D}_g \neq \mathcal{D}_{g'}, \forall g, g' \in [G], g \neq g'$.*

Assumption 2 *The norm of every input sample $\mathbf{x}$ with label y in the Hilbert space i.i.d. drawn from the distribution $\mathcal{D}_G$ of the online interval is upper bounded by a constant D:*

$$\|\mathbf{x}\| \leq D, \forall (\mathbf{x}, y) \sim \mathcal{D}_G.$$

The eigendecomposition of the Hilbert-Schmidt operator is

$$\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_G}[\mathbf{x}\mathbf{x}^T] = \sum_{i=1}^{\infty} \lambda_i \mathbf{u}_i \mathbf{u}_i^T,$$

where $(\mathbf{u}_i)_{i=1}^{\infty}$ forms an orthonormal basis of Hilbert Space and $(\lambda_i)_{i=1}^{\infty}$ corresponds to the eigenvalues in a non-increasing order.

Assumption 3 *For any sample $(\mathbf{x}, y) \sim \mathcal{D}_{\mathcal{U}}$, the hypothesis class is*

$$\mathcal{H} \triangleq \{h : \mathbf{x} \mapsto \langle \mathbf{w}, \mathbf{x} \rangle \mid \mathbf{w} \in \mathcal{W}, \|\mathbf{w}\| \leq R\},$$

where the domain $\mathcal{W}$ bounded by R is a convex subspace of a Hilbert space.

Assumption 4 *For any sample $(\mathbf{x}, y) \sim \mathcal{D}_{\mathcal{U}}$, the loss function family $\mathcal{L}$ with the hypothesis class $\mathcal{H}$ is bounded in the interval $[0, 1]$:*

$$\mathcal{L} \triangleq \{(\mathbf{x}, y) \mapsto l(h(\mathbf{x}), y) \mid h \in \mathcal{H}, l(h(\mathbf{x}), y) \in [0, 1]\}.$$

Assumption 5 *For any $(\mathbf{x}, y) \sim \mathcal{D}_{\mathcal{U}}$ and all $\mathbf{w}, \mathbf{w}' \in \mathcal{W}$, $l(\langle \cdot, \mathbf{x} \rangle, y)$ is convex and β-smooth over the domain $\mathcal{W}$:*

$$\|\nabla l(\langle \mathbf{w}, \mathbf{x} \rangle, y) - \nabla l(\langle \mathbf{w}', \mathbf{x} \rangle, y)\| \leq \beta \|\mathbf{w} - \mathbf{w}'\|.$$

In the G^{th} interval, we would like to learn an expert $\mathbf{w} \in \mathcal{W}$ with a small popular risk with respect to the nonnegative loss function l

$$L_{\mathcal{D}}(\mathbf{w}) = \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}}[l(\langle \mathbf{w}, \mathbf{x} \rangle, y)], \tag{1}$$

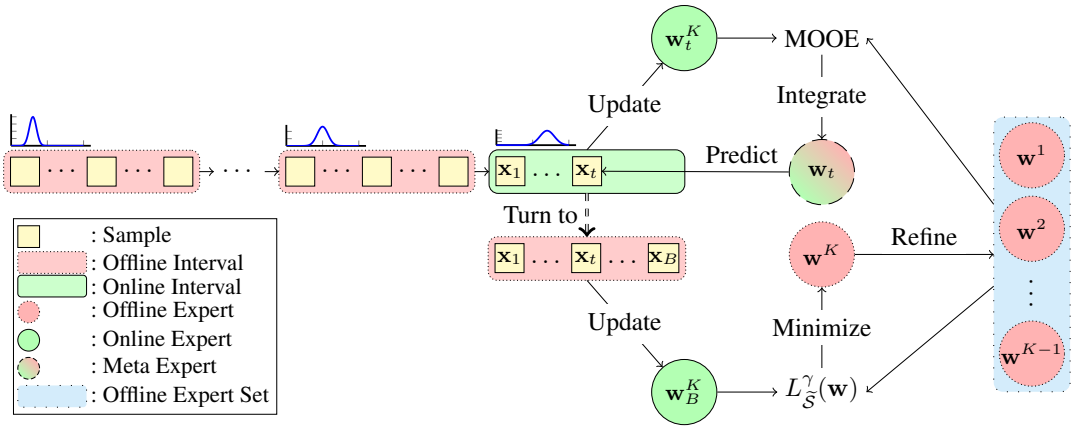

Figure 1: The working process of MOOE on non-stationary time series.

by minimizing the corresponding empirical risk using the proposed method:

$$L_{\mathcal{S}}(\mathbf{w}) = \frac{1}{T}\sum_{t=1}^{T} l(\langle \mathbf{w}, \mathbf{x}_t \rangle, y_t) = \frac{1}{T}\sum_{t=1}^{T} f_t(\mathbf{w}), \qquad (2)$$

where $\mathcal{S} = \{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_T, y_T)\}$ is the data set consisting of $T(T \in [B])$ samples in the online interval, and we use $L_{\widetilde{\mathcal{S}}}(\mathbf{w})$ to denote the specific case when $T = B$. Let $\mathbf{w}^* \in \arg\min_{w\in\mathcal{W}} L_{\mathcal{D}}(\mathbf{w})$ be an optimal solution and $\widehat{\mathbf{w}} \in \arg\min_{w\in\mathcal{W}} L_{\mathcal{S}}(\mathbf{w})$ be an empirical minimizer.

Because the loss function l is nonnegative as well as β-smooth, according to the self-bounding property (Srebro, Sridharan, and Tewari 2010) of smooth functions and Assumption 4, we obtain the following upper bound on the norm of the gradients of $l(\langle \cdot, \mathbf{x} \rangle, y)$ for any $(\mathbf{x}, y) \sim \mathcal{D}_{\mathcal{U}}$ and all $\mathbf{w} \in \mathcal{W}$:

$$\|\nabla l(\langle \mathbf{w}, \mathbf{x} \rangle, y)\| \leq \sqrt{4\beta \cdot l(\langle \mathbf{w}, \mathbf{x} \rangle, y)} \leq 2\sqrt{\beta}. \qquad (3)$$

Note that this paper aims to address the non-stationary issue rather than the widely-explored non-convex problem. We thus assume the loss function is convex for convenience and focus on providing the theoretical guarantees for the proposed learning mechanism.

Mixture of Online and Offline Experts

Fig. 1 introduces the working process of Mixture of Online and Offline Experts (MOOE) for the non-stationary time series. MOOE maintains several offline experts for the corresponding offline intervals and an online expert for the current online interval. It then integrates all of these experts using a meta expert with adaptive weights.

The number of maintained experts is K, which is defined as,

$$K = \begin{cases} K_{\max}, & \text{if } G \geq K_{\max} \\ G, & \text{if } G < K_{\max} \end{cases}. \qquad (4)$$

where $K_{\max}$ is a hyperparameter denoting the maximal number of maintained experts. Therefore, MOOE contains $K-1$ offline experts and one online expert. In the interest of brevity, an expert and its corresponding advice are denoted as its parameters $\mathbf{w}$. Accordingly, we assume the k^{th} $(k \in [K-1])$ offline expert is $\mathbf{w}_t^k$ and the online expert is $\mathbf{w}_t^K$. For the t^{th} sample with feedback in the online interval, MOOE firstly selects K experts

$$\{\underbrace{\mathbf{w}_t^1, \ldots, \mathbf{w}_t^{K-1}}_{\text{Offline Expert}}, \quad \underbrace{\mathbf{w}_t^K}_{\text{Online Expert}} \quad\}, \qquad (5)$$

and integrates them into a meta expert $\mathbf{w}_t$ for making a prediction. When $T = B$, the online interval becomes offline, and a new online interval appears. We generate the new offline expert $\mathbf{w}^K$ for the just-passed complete online interval and refresh $K-1$ offline experts if $G \geq K_{\max}$.

Meta Expert

The meta expert adjusts its strategy of integrating the K experts ($K-1$ offline experts and one online expert) according to their losses received on labeled samples. For the online interval, we track the best expert (Herbster and Warmuth 1995) based on the exponentially weighted average forecaster (Cesa-Bianchi and Lugosi 2006) by assigning a considerable weight to the expert with a small cumulative loss, and vice verse. Accordingly, at iteration t in the online interval, the meta expert outputs a weighted average solution

$$\mathbf{w}_t = \sum_{k=1}^{K-1} \alpha_t^k \mathbf{w}_t^k + \alpha_t^K \mathbf{w}_t^K = \sum_{k=1}^{K} \alpha_t^k \mathbf{w}_t^k, \qquad (6)$$

α_t^k is the weight of the k^{th} expert $\mathbf{w}_t^k$. To lead to a compact regret bound, ensure that $\sum_{k=1}^{K} \alpha_1^k = 1$, and provide different weights for experts according to their priorities, α_t^k is initialized as

$$\alpha_1^k = \frac{K+1}{(K+1-k)(K+2-k)K}. \qquad (7)$$

Note that it is unnecessary to project $\mathbf{w}_t$ into the domain $\mathcal{W}$. Because each expert satisfies $\mathbf{w}_t^k \in \mathcal{W}(k \in [K])$ and the weighting function Eq. (6) is linear, the weighted average $\mathbf{w}_t$ is still in the domain $\mathcal{W}$ according to convex properties.

Algorithm 1: MOOE

Input: step size ν
online expert $\mathbf{w}_1^K$
offline expert set $\{\mathbf{w}^1, \ldots, \mathbf{w}^{K-1}\}$
Initialize $\alpha_1^1 < \alpha_1^2 < \cdots < \alpha_1^K$ according to:

$$\alpha_1^k = \frac{K+1}{(K+1-k)(K+2-k)K}, \forall k \in [K]$$

for t = 1,...,T **do**
  Receive online expert $\mathbf{w}_t^K$
  Assign offline expert $\mathbf{w}_t^k = \mathbf{w}^k, \forall k \in [K-1]$
  Output weighted average: $\mathbf{w}_t = \sum_{k=1}^K \alpha_t^k \mathbf{w}_t^k$
  Receive the loss function $f_t(\cdot)$
  Update expert weights:

$$\alpha_t^k = \frac{\alpha_t^k e^{-\nu f_t(\mathbf{w}_t^k)}}{\sum_{k'=1}^K \alpha_t^{k'} e^{-\nu f_t(\mathbf{w}_t^{k'})}}, \forall k \in [K]$$

  Send gradient $\nabla f_t(\mathbf{w}_t^K)$ to the online expert
end for

After obtaining the loss at iteration t, the K weights are updated according to the exponential weighting scheme

$$\alpha_t^k = \frac{\alpha_t^k e^{-\nu f_t(\mathbf{w}_t^k)}}{\sum_{k'=1}^K \alpha_t^{k'} e^{-\nu f_t(\mathbf{w}_t^{k'})}}, \tag{8}$$

where $\nu = 4\sqrt{\frac{\ln K}{T}}$ is the step size. MOOE is summarized in Algorithm 1.

Offline Expert

We extract knowledge from offline intervals by learning an offline expert for each online interval when all of its samples are available. Each interval is coupled with its previous offline experts and online expert when its online expert has passed this interval once, and its previous offline experts may be learned from similar distributions. Thus, we transfer their knowledge adaptively to its offline expert.

For each online interval, we calculate a new offline expert $\mathbf{w}^K$ once $T = B$ by taking advantage of the prior knowledge of the $K-1$ offline experts $\mathbf{w}_B^1, \ldots, \mathbf{w}_B^{K-1}$ and the online expert $\mathbf{w}_B^K$. According to the strategy of the meta expert, the expert performing best in this interval has the largest weight, therefore the new offline expert $\mathbf{w}^K$ should be close to the best expert. Accordingly, we use the regularization term $\Omega(\mathbf{w}) = \left\|\mathbf{w} - \sum_{k=1}^K \alpha_B^k \mathbf{w}_B^k\right\|_2^2$ to constrain the search space of $\mathbf{w}^K$,

$$\mathbf{w}^K = \arg\min_{w \in \mathcal{W}} \underbrace{\frac{1}{B}\sum_{t=1}^B f_t(\mathbf{w}) + \frac{\gamma}{2}\Omega(\mathbf{w})}_{L_{\widetilde{\mathcal{S}}}^{\gamma}(\mathbf{w})}, \tag{9}$$

where $\gamma \geq \sum_{k=1}^K \alpha_B^k L_{\widetilde{\mathcal{S}}}(\mathbf{w}_B^k)/4R^2$ is a hyperparameter to control the effect of the prior knowledge and T is assigned as B in $\widetilde{\mathcal{S}}$.

After receiving the new offline expert $\mathbf{w}^K$, we set priorities for all K offline experts, as their potential abilities for the next online interval vary. We then select $K-1$ offline experts by eliminating the one with the lowest priority and initialize their weights in the meta expert according to these priorities, as shown in Eq. (7). The priority mechanism does not affect our theoretical results for the MOOE method, so we do not delve into it here. Instead, we provide two simple mechanisms: maintaining an expert queue where the first expert has the lowest priority and the newest expert is enqueued while the oldest is removed, or setting the newest offline expert with the highest priority and assigning the priorities for the previous $K-1$ experts based on their weights.

Online Expert

To train an online expert for a new online interval, we can reinitialize its parameters randomly or by inheriting its solution from the just-passed complete online interval as a warm start. Recall that we can train the online expert by any off-the-shelf online optimization methods on the fly. In this paper, we use the standard Online Gradient Descent (OGD) (Zinkevich 2003) method as an instance because it is the most common and famous online optimization method. On the online interval, the online expert submits its advice $\mathbf{w}_t^K$ to the meta expert and receives the gradient $\nabla f_t(\mathbf{w}_t^K)$ to update its parameters by

$$\mathbf{w}_{t+1}^K = \Pi_{\mathcal{W}}[\mathbf{w}_t^K - \eta_t \nabla f_t(\mathbf{w}_t^K)] \tag{10}$$

$\eta_t = \frac{D}{\sqrt{\beta t}}$ is the step size, and $\Pi_{\mathcal{W}}$ is the proximal operator onto space $\mathcal{W}$.

Theoretical Guarantees

In this section, we provide theoretical guarantees for MOOE, which match our expectations. Specifically, we analyze the properties of the regularization term $\Omega(\mathbf{w})$ and provide the regret and the generalization error of the output hypothesis. To exploit the convexity, smoothness, and nonnegativity conditions of the loss function, the hypothesis class, the data distribution, and the regret, we involve the data-independent excess risk of $\widehat{\mathbf{w}}$, the Rademacher complexity of hypothesis class $\mathcal{H}$ w.r.t. $\mathcal{D}$ and the regret for implying the generalization. Detailed theoretical derivations are provided in the full version of the paper (Zhao, Cao, and Wan 2024).

Parameter Convergence

The hyperparameter γ for $\Omega(\mathbf{w})$ should be assigned with considerable value to ensure the validity of the regularization. To process, we derive the upper bound of this regularization.

Lemma 1 *$L_{\widetilde{\mathcal{S}}}^{\gamma}(\mathbf{w})$ is strongly-convex w.r.t. $\mathbf{w} \in \mathcal{W}$, and*

$$\Omega(\mathbf{w}^K) \leq \sum_{k=1}^K \alpha_T^k L_{\widetilde{\mathcal{S}}}(\mathbf{w}_B^k)/\gamma.$$

We set $\gamma \geq \left(\sum_{k=1}^K \alpha_B^k L_{\widetilde{\mathcal{S}}}(\mathbf{w}_B^k)\right)/4R^2$ to ensure the validity of the regularization term.

Accordingly, the following theorem shows the benefit of this regularization, it can narrow the gap between the minimizer $\mathbf{w}^K$ and the optimal solution $\mathbf{w}^*$ by applying the maintained K experts adaptively and setting γ carefully.

Theorem 1 *By setting* $\gamma \geq \left(\sum_{k=1}^{K} \alpha_B^k L_{\widetilde{\mathcal{S}}}(\mathbf{w}_B^k)\right)/4R^2$ *and using* $\mathbf{w}_B^1, \mathbf{w}_B^2, \ldots, \mathbf{w}_B^K$ *as prior knowledge to obtain* $\mathbf{w}^K$ *from* $L_{\widetilde{\mathcal{S}}}^{\gamma}(\mathbf{w})$*, we have*

$$\left\|\mathbf{w}^K - \mathbf{w}^*\right\| \leq \sqrt{2\Omega(\mathbf{w}^*) + \frac{32\beta}{\gamma^2} + \frac{6\sum_{k=1}^{K} \alpha_B^k L_{\widetilde{\mathcal{S}}}(\mathbf{w}_B^k)}{\gamma}}.$$

Although it is impossible for us to obtain $\mathbf{w}^*$ since the distribution for this interval is unknown, we can obtain an approximate solution by using the regularization term $\Omega(\mathbf{w})$. If the optimal solution is close to the weight average $\sum_{k=1}^{K} \alpha_T^k \mathbf{w}_T^k$, the value of $\Omega(\mathbf{w}^*)$ and the upper bound of the difference $\left\|\mathbf{w}^K - \mathbf{w}^*\right\|$ are small. Although it is also impossible for us to measure $\Omega(\mathbf{w}^*)$, we can measure the weighted term $\sum_{k=1}^{K} \alpha_B^k L_{\widetilde{\mathcal{S}}}(\mathbf{w}_B^k)$ in the above upper bound where $L_{\widetilde{\mathcal{S}}}(\mathbf{w}_B^k)$ is the empirical error of the k^{th} expert in the latest interval. As a result, we know that the empirical minimizer $\mathbf{w}^K$ of $L_{\widetilde{\mathcal{S}}}^{\gamma}(\mathbf{w})$ approaches the optimal solution $\mathbf{w}^*$ of the original problem $L_{\mathcal{D}}(\mathbf{w})$ if these experts considered in the regularization term $\Omega(\mathbf{w})$ are effective in the latest interval. To sharpen this bound, the weights for experts with small empirical errors should be larger and the design of the meta expert can meet the need. Therefore, we can draw a conclusion that $\mathbf{w}^K$ should be close to these experts with small empirical errors in the domain $\mathcal{W}$. This conclusion leads to the design of the regularization term $\Omega(\mathbf{w})$.

Regret Bound

The following regret measures the performance of MOOE

$$\text{Regret}_{\text{MOOE}} = \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{\mathbf{w}\in\mathcal{W}} \sum_{t=1}^{T} f_t(\mathbf{w}). \tag{11}$$

However, it is hard to minimize the regret directly because the output $\mathbf{w}_t$ is related to a meta expert, an online expert, and $K-1$ offline experts. Therefore, we decompose the regret into two regrets: $\text{Regret}_{\text{ME}}$ w.r.t. the meta expert and $\text{Regret}_{\text{KE}}$ w.r.t. online and offline experts. Further, we can bound $\text{Regret}_{\text{KE}}$ by $\text{Regret}_{\text{OE}}$ which corresponds to the online expert. Therefore, we can obtain the regret bound of $\text{Regret}_{\text{MOOE}}$ by bounding $\text{Regret}_{\text{ME}}$ and $\text{Regret}_{\text{OE}}$ separately.

$$\begin{aligned} \text{Regret}_{\text{MOOE}} &= \text{Regret}_{\text{ME}} + \text{Regret}_{\text{KE}} \\ &\leq \text{Regret}_{\text{ME}} + \text{Regret}_{\text{OE}}, \end{aligned} \tag{12}$$

where

$$\begin{aligned} \text{Regret}_{\text{ME}} &= \sum_{t=1}^{T} f_t(\mathbf{w}_t) - \min_{k\in[K]} \sum_{t=1}^{T} f_t(\mathbf{w}_t^k), \\ \text{Regret}_{\text{KE}} &= \min_{k\in[K]} \sum_{t=1}^{T} f_t(\mathbf{w}_t^k) - \sum_{t=1}^{T} f_t(\widehat{\mathbf{w}}), \\ \text{Regret}_{\text{OE}} &= \sum_{t=1}^{T} f_t(\mathbf{w}_t^K) - \sum_{t=1}^{T} f_t(\widehat{\mathbf{w}}). \end{aligned} \tag{13}$$

The online expert $\mathbf{w}_t^K$ never surpasses that of the best expert among all the K experts because it is also one of them. Besides, it is impossible to obtain the regret for the offline experts since they are pre-given and their parameters do not change after receiving the loss $f_t(\cdot)$. Specifically, we have the following theorem.

Theorem 2 *The MOOE method with step sizes* $\{\nu = 4\sqrt{\frac{\ln K}{T}}, \eta_t = \frac{D}{\sqrt{\beta t}}, t \in [T]\}$ *guarantees the following regret for all* $1 \leq T \leq B$*,*

$$\textit{Regret}_{\textit{MOOE}} \leq \sqrt{T \ln K} + 6D\sqrt{T\beta},$$

and the number of experts K *and samples* T *should satisfy*

$$K \leq 2\exp\left(6D\sqrt{\beta} - \frac{\textit{Regret}_{\textit{KE}}}{\sqrt{T}}\right),$$

to ensure that the advice from MOOE gives an equivalent or better result than that from its online expert.

Accordingly, the regret of MOOE for the online interval is $O(\sqrt{T})$, which is consistent with that of the chosen online expert. However, MOOE works better, i.e. $\text{Regret}_{\text{MOOE}} \leq \text{Regret}_{\text{OE}}$, if K and T satisfy the condition in Theorem 2. In theory, we have $\text{Regret}_{\text{KE}} \leq \text{Regret}_{\text{OE}} \leq 6D\sqrt{T\beta}$. These offline experts are better than the online expert when their corresponding data distributions are approximately matched, or the number of observed labeled samples in the current interval is limited. The first inequality is strict, and K is bounded by a positive value. On the other hand, the number of samples in an interval $T \leq B$ should not be too large. Although the bound of K depends on $\text{Regret}_{\text{KE}}$, it is impossible to bound this term without any further assumptions because the K experts are trained from different data sets. Fortunately, it is unnecessary to set K strictly according to its conditions. We can apply MOOE if we believe that the regret of the best offline expert can surpass that of the online expert at least $6D\sqrt{T\beta} - \text{Regret}_{\text{KE}} = \sqrt{T \ln K}$. The assumption is mild since we can set a small K (like 2 or 3) even without prior knowledge. An intuitive understanding is that: if K is too large, it is difficult for the meta expert to derive effective advice because of the dilution effect from those weak experts; if B is too large, the samples in an interval may come from various distributions, and the assumption about the setting may not hold.

Generalization Error Bound

The MOOE performance is measured by the excess risk $L_{\mathcal{D}}(\overline{\mathbf{w}}) - L_{\mathcal{D}}(\mathbf{w}^*)$ where $\overline{\mathbf{w}} = \frac{1}{T}\sum_{t=1}^{T} \mathbf{w}_t$ is the average of the online interval. To derive an algorithmic bound, we introduce the intermediate term $L_{\mathcal{D}}(\widehat{\mathbf{w}})$ because $\widehat{\mathbf{w}}$ as an empirical minimizer of $L_{\mathcal{S}}(\widehat{\mathbf{w}})$ is necessary for analyzing the regret. Taking the divide-and-conquer approach, we have

$$\begin{aligned} &L_{\mathcal{D}}(\overline{\mathbf{w}}) - L_{\mathcal{D}}(\mathbf{w}^*) \leq \frac{1}{T}\sum_{t=1}^{T} L_{\mathcal{D}}(\mathbf{w}_t) - L_{\mathcal{D}}(\mathbf{w}^*) = \\ &\underbrace{\frac{1}{T}\sum_{t=1}^{T} L_{\mathcal{D}}(\mathbf{w}_t) - L_{\mathcal{D}}(\widehat{\mathbf{w}})}_{\mathcal{B}_1} + \underbrace{L_{\mathcal{D}}(\widehat{\mathbf{w}}) - L_{\mathcal{D}}(\mathbf{w}^*)}_{\mathcal{B}_2}. \end{aligned} \tag{14}$$

The inequality is owing to the convexity of $L_{\mathcal{D}}(\cdot)$, which implies $L_{\mathcal{D}}(\frac{1}{T}\sum_{t=1}^{T}\mathbf{w}_t) \leq \frac{1}{T}\sum_{t=1}^{T} L_{\mathcal{D}}(\mathbf{w}_t)$. The regret of MOOE is applied to imply the upper bound of $\mathcal{B}_1$ by the following lemma.

Lemma 2 *Following Theorem 2, with probability at least* $1-\delta$*, we have*

$$\frac{1}{T}\sum_{t=1}^{T} L_{\mathcal{D}}(\mathbf{w}_t) - L_{\mathcal{D}}(\widehat{\mathbf{w}}) \leq \frac{\sqrt{\ln K} + 6D\sqrt{\beta} + 4\log(4/\delta)}{\sqrt{T}}.$$

Following the advanced study for any norm-regularized hypothesis class (Yousefi et al. 2018) and the self-bound property of smooth functions (Srebro, Sridharan, and Tewari 2010), we can derive the following data-dependent generalization bound for $\mathcal{B}_2$ by the following lemmas.

Lemma 3 *Exploiting the convexity, smoothness, and nonnegativity conditions of the loss function family* $\mathcal{L}$*, with probability at least* $1-\delta$*,* $L_{\mathcal{D}}(\widehat{\mathbf{w}}) - L_{\mathcal{D}}(\mathbf{w}^*)$ *is bounded by*

$$\frac{\left(12\beta R^2 + 4R\sqrt{\beta}\right)\log(4/\delta)}{T} + 4R\sqrt{\frac{2\beta\log(4/\delta)}{T}}.$$

Lemma 4 *Exploiting the hypothesis class* $\mathcal{H}$ *and the distribution* $\mathcal{D}$ *of the observed data at the online interval, with probability at least* $1-\delta$*,* $L_{\mathcal{D}}(\widehat{\mathbf{w}}) - L_{\mathcal{D}}(\mathbf{w}^*)$ *is bounded by*

$$42\sqrt{6\beta}\log^{\frac{3}{2}}(64T)\mathcal{R}_{\mathcal{D}}(\mathcal{H}) + 3\sqrt{\frac{\log(4/\delta)}{T}}.$$

where $\mathcal{R}_{\mathcal{D}}(\mathcal{H})$ is the Rademacher complexity of hypothesis space $\mathcal{H}$.

Using the excess risk bound framework in Eq. (14), we obtain the following generalization error bound by considering Lemma 2, Lemma 3 and Lemma 4.

Theorem 3 *Exploiting the loss function properties (convexity, smoothness, and nonnegativity) of* $\mathcal{L}$*, the hypothesis class* $\mathcal{H}$*, the data distribution* $\mathcal{D}$ *and the regret of MOOE, with probability at least* $1-\delta$*, we have*

$$L_{\mathcal{D}}(\overline{\mathbf{w}}) - L_{\mathcal{D}}(\mathbf{w}^*) \leq \frac{\left(12\beta R^2 + 4R\sqrt{\beta}\right)\log(16/\delta)}{T}$$
$$+ \frac{28R\sqrt{\beta}\log^{\frac{3}{2}}(64T)}{T}\left(\sqrt{\sum_{i=1}^{\infty}(TD^2 \wedge e\lambda_i)} + D\sqrt{e}\right)$$
$$+ \frac{\left(6(R+D)\sqrt{\beta}+2\right)\sqrt{\log(16/\delta)} + 4\log(8/\delta) + \sqrt{\ln K}}{\sqrt{T}}.$$

The convergence rate for the generalization error is $O(1/\sqrt{T})$, which is consistent with that in stationary and non-algorithmic cases (Kakade, Sridharan, and Tewari 2008). The bound reflects the best result achieved so far without any other assumptions. Furthermore, it is directly related to the sample complexity, and the result is algorithmic. This result triggers an immediate problem: Can we use fewer samples to achieve a desirable generalization error if the used off-the-shelf online optimization method can achieve a better regret? Unfortunately, the answer is not affirmative. The intuition behind the problem is that the bottleneck is not on the optimization method.

Dataset	NSE	DTEL	Condor	MOOE
Usenet	63.8	68.0	73.1	**78.5**
Weather	76.0	68.9	79.4	**82.4**
GasSensor	42.4	63.8	81.6	**83.1**
Powersupply	74.0	69.9	72.8	**77.9**
Electricity	79.0	81.0	84.7	**85.6**
Covertype	79.0	69.4	89.6	**90.0**
WESAD	70.4	73.8	86.3	**89.9**
Kitsune	73.9	71.6	87.3	**93.4**

Table 1: Comparisons on real-world data.

Experimental Results

In this section, we present empirical analysis to support our proposed theory and model[1].

Regret on Synthetic and Real-World Datasets

We address binary classification on non-stationary time series and compare MOOE with OGD using both synthetic and real-world datasets (ijcnn and cod-rna) from the LIBSVM repository (Chang and Lin 2011). On the synthetic dataset, each interval features samples from two-dimensional Gaussian distributions with dynamically changing means. For the synthetic datasets, we divide the data into 15 intervals, applying Gaussian noise to simulate dynamic changes. We maintain a maximum of five experts in MOOE to ensure fairness in comparison. Theoretical analyses show that MOOE outperforms OGD in dynamic environments, maintaining a convergence rate of $O(1/\sqrt{T})$. As shown in Fig. 2, Empirical results indicate that MOOE achieves significantly lower loss than OGD, particularly at the early stages with few samples, due to the integration of offline expert knowledge. Additionally, MOOE exhibits smaller regret over time, adapting effectively to samples by adjusting the strategy of integrating offline and online experts.

Predictive Accuracy

Real-World Non-Stationary Time Series To verify the effect of the proposed MOOE method, we perform comparison experiments following the setup (Zhao, Cai, and Zhou 2018). Specifically, we use eight real-world non-stationary time series datasets, including Usenet (Katakis, Tsoumakas, and Vlahavas 2008), Weather (Elwell and Polikar 2011), GasSensor (Vergara et al. 2012), Powersupply (Dau et al. 2019), Electricity (Harries 1999) ,Covertype (Sun et al. 2018), WESAD (Schmidt et al. 2018), and Kitsune (Mirsky et al. 2018). We compare MOOE with three state-of-the-art methods, including NSE (Elwell and Polikar 2011), DTEL (Sun et al. 2018), and Condor (Zhao, Cai, and Zhou 2018). In the experiments, we adopt the maximum sample size of an interval $B = 50$ and the maximal number of maintained experts $K_{\max} = 25$. The overall mean of predictive accuracy is reported, which indicates the average performance of the algorithm over the whole time series. The com-

[1]The source code is publicly available at: https://github.com/Lawliet-zzl/MOOE.

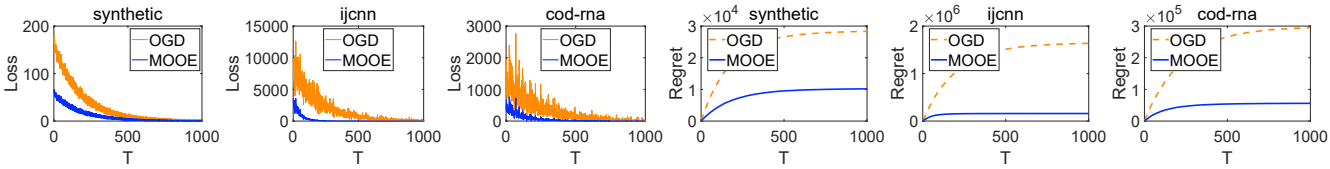

Figure 2: Regret and loss of MOOE and OGD methods.

Algorithm	Email list			Spam filtering		
	Accuracy	Precision	Recall	Accuracy	Precision	Recall
NSE	70.0	76.5	76.5	90.4	84.5	79.6
DTEL	86.2	88.2	88.2	86.3	73.4	71.4
Condor	95.6	93.2	**99.8**	95.4	91.1	90.8
MOOE	**97.1**	**94.0**	**99.8**	**96.0**	**93.4**	**92.8**

Table 2: Comparisons on data with recurring concept drift.

Approach	NSE	DTEL	Condor	MOOE
Accuracy	64.9	58.7	80.1	**86.5**

Table 3: Comparisons on data with increasing noise.

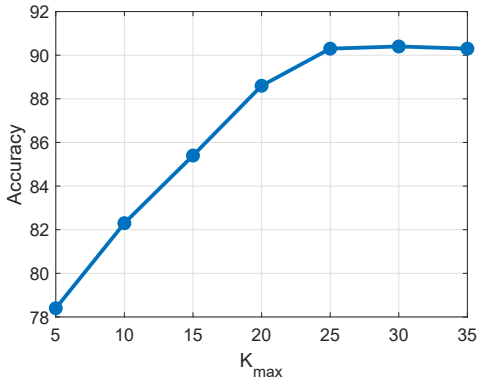

Figure 3: MOOE with different $K_{\max}$.

parison results are reported in Table 1. The results show that the proposed MOOE method outperforms other contenders. Specifically, MOOE achieves 13.7% improvement over the other state-of-the-art methods. This is because MOOE can utilize the knowledge of the offline experts to adopt each new online interval. These experimental results show the superiority of the proposed MOOE method.

Non-stationary Time Series with Recurring Concept Drift To verify the versatility of the proposed MOOE method, we conduct the comparisons on a special case of non-stationary time series, i.e., recurring concept drift, in which previous distributions may disappear and then re-appear in the future. We consider two real-world non-stationary time series with recurring concept drift, namely Email list and Spam filtering (Katakis, Tsoumakas, and Vlahavas 2010). The concepts are decided by the personal interests of users that change in a recurring manner. The results are summarized in Table 2, which show that MOOE exhibits an encouraging performance on the two datasets regarding all measures. Specifically, MOOE achieves 11.8% improvement in terms of accuracy. This is because the offline experts can learn the knowledge on the previous distributions, and the meta expert can reuse the knowledge when the distribution re-appear.

Non-Stationary Time Series with Increasing Levels of Noise To verify the robustness of the proposed MOOE method, we perform experiments on non-stationary time series with increasing levels of noise. Specifically, we adopt Covertype and gradually add Gaussian noise until the time series becomes completely random. The experiment results presented in Table 3 indicate that MOOE achieves the best prediction result. This is because MOOE can utilize the knowledge learned by the offline experts when the online expert is hard to learn knowledge from the noisy data.

Effect of Maximal Number of Maintained Experts

To verify the effect of the hyper-parameter $K_{\max}$, we select its value from $\{5, 10, 15, 20, 25, 30, 35\}$ and perform experiments on Covertype. The experimental results presented in Fig. 3 show that increasing $K_{\max}$ can improve the classification performance and the performance stabilizes when $K_{\max}$ is sufficiently large, e.g., $K_{\max} = 25$. This is because a large number of maintained experts $K_{\max}$ causes more knowledge can be stored and applied for the data in the online interval. Furthermore, if $K_{\max}$ is sufficiently large, increasing its number cannot make MOOE obtain the new knowledge that the offline experts have not explored.

Conclusion

In this paper, we address a general and realistic scenario involving non-stationary time series, where several offline intervals with various distributions exist alongside an online interval. We propose MOOE, which employs a meta expert to integrate static offline experts, learned from previous offline intervals, with the dynamic online expert, updated in the online interval. We provide theoretical guarantees regarding parameter convergence, regret bounds, and generalization error bounds. Our theoretical results demonstrate that MOOE achieves the same generalization error bounds in both stationary and non-stationary cases, proving that leveraging knowledge from historical intervals is effective. Future work will explore other assumptions and techniques to overcome bottlenecks in the generalization bound.

Acknowledgments

This work was supported by the Australian Research Council through the Linkage Grant (LP230201022), the Discovery Grant (DP240102050), and the Linkage Infrastructure, Equipment, and Facilities Grant (LE240100131).

References

Adams, T. M.; and Nobel, A. B. 2010. Uniform convergence of Vapnik-Chervonenkis classes under ergodic sampling. *The Annals of Probability*, 38(4): 1345–1367.

Agarwal, A.; and Duchi, J. C. 2013. The Generalization Ability of Online Algorithms for Dependent Data. *IEEE Trans. Inf. Theory*, 59(1): 573–587.

Anagnostides, I.; Farina, G.; and Sandholm, T. 2023. Near-Optimal Φ-Regret Learning in Extensive-Form Games. In Krause, A.; Brunskill, E.; Cho, K.; Engelhardt, B.; Sabato, S.; and Scarlett, J., eds., *ICML*, 814–839.

Besbes, O.; Gur, Y.; and Zeevi, A. J. 2015. Non-stationary Stochastic Optimization. *Operations Research*, 63(5): 1227–1244.

Buchbinder, N.; Chen, S.; Naor, J.; and Shamir, O. 2016. Unified Algorithms for Online Learning and Competitive Analysis. *Math. Oper. Res.*, 41(2): 612–625.

Cesa-Bianchi, N.; and Lugosi, G. 2006. *Prediction, Learning, and Games*. Cambridge University Press.

Chang, C.; and Lin, C. 2011. LIBSVM: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3): 1–27.

Dau, H. A.; Bagnall, A.; Kamgar, K.; Yeh, C.-C. M.; Zhu, Y.; Gharghabi, S.; Ratanamahatana, C. A.; and Keogh, E. 2019. The UCR Time Series Archive.

Elwell, R.; and Polikar, R. 2011. Incremental Learning of Concept Drift in Nonstationary Environments. *IEEE Trans. Neural Networks*, 22(10): 1517–1531.

Hamilton, J. D. 1994. *Time Series Analysis*. Princeton.

Hanneke, S. 2016. The Optimal Sample Complexity of PAC Learning. *J. Mach. Learn. Res.*, 17(38): 1–15.

Harries, M. B. 1999. SPLICE-2 Comparative Evaluation: Electricity Pricing. In *Technical Report of South Wales University*.

Hazan, E. 2016. Introduction to Online Convex Optimization. *Foundations and Trends in Optimization*, 2(3-4): 157–325.

Herbster, M.; and Warmuth, M. K. 1995. Tracking the Best Expert. In *ICML*, 286–294.

Kakade, S. M.; Sridharan, K.; and Tewari, A. 2008. On the Complexity of Linear Prediction: Risk Bounds, Margin Bounds, and Regularization. In *NeurIPS*, 793–800.

Katakis, I.; Tsoumakas, G.; and Vlahavas, I. P. 2008. An Ensemble of Classifiers for coping with Recurring Contexts in Data Streams. In *ECAI*, 763–764.

Katakis, I.; Tsoumakas, G.; and Vlahavas, I. P. 2010. Tracking recurring contexts using ensemble classifiers: an application to email filtering. *Knowl. Inf. Syst.*, 22(3): 371–391.

Kuznetsov, V.; and Mohri, M. 2014. Generalization Bounds for Time Series Prediction with Non-stationary Processes. In *ALT*, 260–274.

Kuznetsov, V.; and Mohri, M. 2015. Learning theory and algorithms for forecasting non-stationary time series. In *NeurIPS*, 541–549.

Mirsky, Y.; Doitshman, T.; Elovici, Y.; and Shabtai, A. 2018. Kitsune: An Ensemble of Autoencoders for Online Network Intrusion Detection. In *NDSS*, 1–15.

Mohri, M.; and Rostamizadeh, A. 2008. Rademacher Complexity Bounds for Non-I.I.D. Processes. In *NeurIPS*, 1097–1104.

Mohri, M.; and Rostamizadeh, A. 2010. Stability Bounds for Stationary phi-mixing and beta-mixing Processes. *J. Mach. Learn. Res.*, 11: 789–814.

Puigcerver, J.; Ruiz, C. R.; Mustafa, B.; and Houlsby, N. 2024. From Sparse to Soft Mixtures of Experts. In *ICLR*.

Schmidt, P.; Reiss, A.; Dürichen, R.; Marberger, C.; and Laerhoven, K. V. 2018. Introducing WESAD, a Multimodal Dataset for Wearable Stress and Affect Detection. In *ICMI*, 1–9.

Shalev-Shwartz, S. 2012. Online Learning and Online Convex Optimization. *Foundations and Trends in Machine Learning*, 4(2): 107–194.

Srebro, N.; Sridharan, K.; and Tewari, A. 2010. Optimistic Rates for Learning with a Smooth Loss. *CoRR*, abs/1009.3896: 1–29.

Sun, Y.; Tang, K.; Zhu, Z.; and Yao, X. 2018. Concept Drift Adaptation by Exploiting Historical Knowledge. *IEEE Trans. Neural Networks Learn. Syst.*, 29(10): 4822–4832.

van Erven, T.; and Koolen, W. M. 2016. MetaGrad: Multiple Learning Rates in Online Learning. In *NeurIPS*, 3666–3674.

Vergara, A.; Vembu, S.; Ayhan, T.; Ryan, M. A.; Homer, M. L.; and Huerta, R. 2012. Chemical gas sensor drift compensation using classifier ensembles. *Sensors and Actuators B: Chemical*, 166-167: 320–329.

Wei, C.; Hong, Y.; and Lu, C. 2019. Tracking the Best Expert in Non-stationary Stochastic Environments. In *NeurIPS*, 3972–3980.

Yousefi, N.; Lei, Y.; Kloft, M.; Mollaghasemi, M.; and Anagnostopoulos, G. C. 2018. Local Rademacher Complexity-based Learning Guarantees for Multi-Task Learning. *J. Mach. Learn. Res.*, 19: 1–47.

Yu, B. 1994. Rates of convergence for empirical processes of stationary mixing sequences. *The Annals of Probability*, 22(1): 94–116.

Zhang, L.; Lu, S.; and Zhou, Z. 2018. Adaptive Online Learning in Dynamic Environments. In *NeurIPS*, 1330–1340.

Zhao, P.; Cai, L.; and Zhou, Z. 2018. Handling concept drift via model reuse. *Mach. Learn.*, 109(3): 533–568.

Zhao, Z.; Cao, L.; and Wan, Y. 2024. Mixture of Online and Offline Experts for Non-stationary Time Series. *CoRR*, abs/2202.05996: 1–14.

Zinkevich, M. 2003. Online convex programming and generalized infinitesimal gradient ascent. In *ICML*, 928–936.